I. REMARKS

A. Examiner Interview Summary

The Applicant interviewed the Examiner by telephone on 8/11/2004.

B. Exhibits & Demonstrations

The Applicant did not present exhibits or conduct demonstrations.

C. Claims Discussed

Claims 12-35 were discussed, particularly independent claims 12, 27, and 31.

D. Substance of Interview

The Applicant submitted a draft of proposed amendments responsive to the office 103(a), as being unpatentable over C. F. Codella et al., *Support for Enterprise JavaBeans in Component Broker*, 37 IBM Systems Journal 4, 502-538 (1998) in view of U.S. Pat. No. 6,529,948 (issued March 4, 2003). The Applicant argued that Codella does not teach all of the limitations of the Applicant's claims, particularly the act of executing an object-oriented query within an EJB container. The Applicant further argued that, rather than disclosing the functionality of the Applicant's invention, Codella merely describes the foundation upon which the present invention builds to enable this functionality. The Examiner and the Applicant discussed Codella in detail, but the Examiner maintained his position with respect to the § 103 rejections. The Examiner further indicated that the proposed amendments would not change his position with respect to Codella.

E. Objections

1. Specification

The Examiner identified a misspelled word ("implemenetation," pg. 17, line 7). The Applicant has amended the specification to overcome the Examiner's objection.

2. Claims

The Examiner objected to the form of claim 31, suggesting that "a readable memory by itself cannot cause to [sic] build a [sic] object." The Applicant, though, respectfully disagrees with the Examiner's interpretation of claim 31. Claim 31 clearly recites structure within the memory (i.e. "a program stored in the storage medium") that causes the computer to execute the steps required for building an object.

F. Claim Rejections

The Examiner rejected all claims per 35 U.S.C. § 103(a), as being obvious in view of C. F. Codella et al., *Support for Enterprise JavaBeans in Component Broker*, 37 IBM Systems Journal 4, 502-538 (1998) and U.S. Pat. No. 6,529,948 (issued March 4, 2003) [hereinafter Bowman].

To establish prima facie obviousness of a claimed invention, though, all the claim limitations must be taught or suggested by the prior art. *See* United States Patent & Trademark Office, *Manual of Patent Examining Procedure* (8th ed. rev. 2 2004) § 2143.03 [hereinafter *MPEP*] (citing *In re Royka*, 490 F.2d 981 (CCPA 1974)). "But most, if not all, inventions arise

invention as a whole. *Id.* at 1369-70; *accord MPEP* §2141.02. "[I]dentification in the prior art of each individual part claimed is insufficient to defeat patentability of the whole claimed invention." *Kotzab*, 217 F.3d at 1370. "Rather, to establish obviousness based on a combination [illegible] of the desirability of making the specific combination . . . ." *Id.*; *accord MPEP* § 2143.01.

As discussed in detail below, neither Codella nor Bowman, in combination or alone, teaches or suggests the claimed invention as a whole. Moreover, even if Codella and Bowman collectively did teach or suggest each individual claim limitation, neither provides any teaching, suggestion, or motivation to modify the prior art to produce the claimed invention as a whole. Thus, the Examiner has not established a prima facie case of obviousness, and the Applicant respectfully requests reconsideration of these rejections.

1. The Prior Art References

The Applicant provides the following summary of the prior art in order to convey the Applicant's current understanding of the reference upon which the Examiner relies.

a. Codella

Codella discusses various component models, which define frameworks for multi-platform, multi-language distributed object interaction. In particular, Codella discusses the Object Management Group's Common Object Request Broker Architecture (CORBA), IBM's Component Broker,[1] and SUN's Enterprise JavaBeans (EJB).

[1] Component Broker is IBM's proprietary CORBA implementation, Codella, *supra*, at 502, and is now part of IBM's Websphere Application Server, Enterprise Edition (WAS/EE), *see, e.g.*, IBM Redbooks, IBM WebSphere Application Server Enterprise Edition Component Broker 3.0: First Steps, at http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg242033.html?Open. (last visited 08/06/2004).

An "object," as that term is used in Codella, encapsulates program code and data, and provides an interface through which a program or other objects within the program can access the data. Codella, *supra*, at 502. In general, an object only "exists" within a single program. *Id.* at 503. But a component model, such as CORBA or Component Broker, provides services that an object can use to communicate and exchange data with any other object that conforms to the component model – even if the other objects exist in another program or an another machine in a network. *Id.*

Codella teaches that Component Broker includes the Managed Object Framework (MOFW) which represents a set of interfaces and conventions that "must be followed in order to specification." *Id.* at 514. *See also id.* at 504 (". . . Component Broker is an example of a CORBA [Object Transaction Monitor] that can be used to deploy EJB components by implementing the contract between a component and its container"). According to Codella, "the main concepts in the EJB specification, namely, Enterprise beans, containers, factories, finders, handles, and so on, correspond in a natural way to concepts in Component Broker such as business objects, homes, instance managers, and son on." *Id.* at 514. Thus, as illustrated in FIG. 8, MOFW adds a layer of "usability and function" to CORBA services, *see id.* at 505, and provides a foundation for implementing an EJB container, *id.* at 514. As Codella further points out, an entity bean "must[2] implement a number of application-specific create and find methods." *Id.* at 521. Container tools (i.e. Component Broker tools) generate an entity bean's finder methods. *Id.*

[2] Presumably, the entity beans "must" implement certain methods to comply with the EJB specification.

b. Bowman

Bowman discusses, among many other things, a method for processing a query for an object stored in a database, so that related objects are returned along with the requested object "in one access operation." Bowman, *supra*, at 2:17-26.

2. The Prior Art Does Not Teach or Suggest All the Limitations of the Claimed Invention

The Examiner states that Codella discloses "a method of building Enterprise Java Bean objects that meet an object-oriented query." (Office Communication from Examiner to Applicant of 7/21/2004, at 4 (discussing independent claim 12) [hereinafter O.A.].) In particular, the Examiner states that Codella, in FIG. 1 and on page 505, discloses the step of "executing an object-oriented query on an application server such that the server returns data objects in response to the query." (*Id.*) The Applicant, though, is unable to identify any subject matter in either FIG. 1 or page 505 that describes the act of executing a query on an application server.

Thus, while Codella teaches many things about implementing an EJB container, including finder methods, Codella does not appear to teach all of the limitations of the Applicant's claims that the Examiner has suggested it teaches. In particular, Codella does not appear to disclose the act of executing an object-oriented query within an EJB container.

3. The Prior Art Does Not Suggest the Desirability of the Claimed Invention

a. The Examiner Did Not Consider the Claimed Invention as a Whole

The Examiner concedes that Codella does not disclose many other elements in data objects, and the conversion of the JAVA Vector to a JAVA Enumeration (*see* O.A., at 4, 5, & 9 (discussing independent claims 12, 27, and 31)), but suggests that Bowman discloses these elements. In particular, the Examiner states that Bowman, "in a method to communicate Corba [sic] environment [sic] with enterprise beans in order to provide object-oriented distribution, discloses [a] server beans implementation of business data and Java Vector in conjunction with [an] Enumeration analogous to the EJB interface by Codella." (O.A., at 5 (citing Bowman, *supra*, at 203:48 to 204:40).)

The cited reference, though, appears to be nothing more than a cursory review of methods for "managing constants in a computer program" (Bowman, *supra*, at 202:33-34), albeit in an "object-based system" (*id.* at 203:15) using a JAVA Vector (*id.* at 204:4) and an Enumeration object (id. at 204:23-35). Again, the Applicant is unable to identify any other subject matter in the cited material that relates to the Applicant's claimed invention as a whole. Particularly within the cited excerpt, the concept of object-oriented queries appears foreign to Bowman, there is no reference to CORBA, as the Examiner has suggested, and the only reference to "enterprise beans" merely indicates that the "example has not yet been updated to JavaBeans" (*id.* at 203:66-

c. The Examiner's reasoning is neither clear nor unmistakable

technical and scientific reasoning to support his or her conclusion of common knowledge." *Id.* (emphasis added) (citing *In re Soli*, 317 F.2d 941, 946 (CCPA 1963); *In re Chevenard*, 139 F.2d 711, 713 (CCPA 1943); *see also Kotzab*, 217 F.3d at 1370 (mandating that the Board of Patent Appeals provide "particular findings" and stating that "[b]road conclusory statements standing alone are not 'evidence'"). The Examiner's underlying technical reasoning of the conclusion based on the "well-known concept," though, is neither "clear" nor "unmistakable." *See id.* The Examiner appears to suggest that the "well-known concept" is either "add[ing] to the creation of EJB objects . . . a Java Vector to contain the objects," using a "Vector to operate with the Enumeration," or "data type safety protection and less type checking burden" (O.A., at 5), but it is unclear if the Examiner is referring to these concepts collectively or individually. If the Examiner is referring to an individual concept, it is not clear to which individual concept the Application served to combine [illegible] as the Applicant has done.

G. Conclusion

The Applicant has specifically pointed out the errors in the Examiner's action and is entitled to documentary evidence or an affidavit that supports the Examiner's reliance on "well-known" concepts. *See MPEP* § 2144.03 C. Without such evidence, the Examiner has neither cited prior art that discloses the Applicant's invention as a whole, nor adequately identified any motivation, suggestion, or teaching of the desirability of the specific combination that the

Applicant claims. Thus, either the Examiner's final rejection was premature, or the Examiner has failed to establish prima facie obviousness of the Applicant's claimed invention.

The Applicant accordingly requests the Examiner to withdraw the finality of the rejections based upon 35 U.S.C. § 103 and provide the Applicant with an opportunity to respond to the Examiner's evidence, *see MPEP* § 2144.03 B, or in the alternative, withdraw the rejections completely and allow the Applicant's claims as presented.